1250428

THE UNITED STATES OF AMERICA

TO ALL TO WHOM THESE PRESENTS SHALL COME:

UNITED STATES DEPARTMENT OF COMMERCE

United States Patent and Trademark Office

November 17, 2004

THIS IS TO CERTIFY THAT ANNEXED HERETO IS A TRUE COPY FROM THE RECORDS OF THE UNITED STATES PATENT AND TRADEMARK OFFICE OF THOSE PAPERS OF THE BELOW IDENTIFIED PATENT APPLICATION THAT MET THE REQUIREMENTS TO BE GRANTED A FILING DATE.

APPLICATION NUMBER: *60/510,068*
FILING DATE: *October 09, 2003*
RELATED PCT APPLICATION NUMBER: *PCT/US04/33290*

Certified by

Jon W Dudas

Acting Under Secretary of Commerce
for Intellectual Property
and Acting Director of the U.S.
Patent and Trademark Office

16519 U.S.PTO

PTO/SB/16 (08-03)
Approved for use through 07/31/2006. OMB 0651-0032
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

22582 U.S.PTO 60/510068 100903

PROVISIONAL APPLICATION FOR PATENT COVER SHEET

This is a request for filing a PROVISIONAL APPLICATION FOR PATENT under 37 CFR 1.53(c).

Express Mail Label No. EV 162112455 US

INVENTOR(S)

Given Name (first and middle [if any])	Family Name or Surname	Residence (City and either State or Foreign Country)
James Edwin	Van Buskirk	Cedar Park, TX

Additional inventors are being named on the ________ *separately numbered sheets attached hereto*

TITLE OF THE INVENTION (500 characters max)

Direct all correspondence to: **CORRESPONDENCE ADDRESS**

☑ Customer Number: 27160

OR

☐ Firm or Individual Name					
Address					
Address					
City		State		Zip	
Country		Telephone		Fax	

ENCLOSED APPLICATION PARTS *(check all that apply)*

☒ Specification *Number of Pages* 15
☒ CD(s), Number 2
☒ Drawing(s) *Number of Sheets* 12
☒ Other (specify) CD TRANSMITTAL SHEET
☐ Application Date Sheet. See 37 CFR 1.76

METHOD OF PAYMENT OF FILING FEES FOR THIS PROVISIONAL APPLICATION FOR PATENT

☐ Applicant claims small entity status. See 37 CFR 1.27.
☐ A check or money order is enclosed to cover the filing fees.
☑ The Director is herby authorized to charge filing fees or credit any overpayment to Deposit Account Number: 50-1214
☐ Payment by credit card. Form PTO-2038 is attached.

FILING FEE Amount ($): $160.00

The invention was made by an agency of the United States Government or under a contract with an agency of the United States Government.

☑ No.
☐ Yes, the name of the U.S. Government agency and the Government contract number are: ________

Respectfully submitted,

SIGNATURE [signature]
Date 09 OCT 03

TYPED or PRINTED NAME John S. Paniaguas
REGISTRATION NO. 31051 *(if appropriate)*
Docket Number: 281450

TELEPHONE 312/902-5312

USE ONLY FOR FILING A PROVISIONAL APPLICATION FOR PATENT

PATENT

Attorney Docket No. 281450/208

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE
PATENT COOPERATION TREATY RECEIVING OFFICE

Applicant(s): TEAC America, Inc.

Serial No.: Herewith

Filing Date: October 9, 2003

Title: Complete Studio

Group Art Unit: N/A

Examiner: N/A

CERTIFICATE OF MAILING BY "EXPRESS MAIL"

"Express Mail" Mailing Label Number
EV 162112455 US

Date of Deposit: October 9, 2003

I hereby certify that this paper or fee is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" Service under 37 CFR §1.10 on the date indicated above and is addressed to the Commissioner for Patents, Washington, D.C. 20231.

Maria Flores
(Typed or printed name of person mailing)

Maria Flores
(Signature of person mailing)

Transmittal Letter for Compact Disc Submission

Enclosed are two copies of one computer program listing. The compact discs are in Read-Only format and are in full compliance with the American Standard Code for Information Interchange (ASCII). The machine format for the compact discs is IBM-PC and the operating system compatibility is Windows. The files and file names contained in the compact disc as well as the file size in bytes are attached hereto. Files with .CPP, .ASM, .H, INC and .TXT are ASCII files and can be opened using Notepad software. The two compact discs are duplicated and are identical.

The Commissioner is hereby authorized to charge any additional fees which may be required in this application under 37 C.F.R. §§1.16-1.17 during its entire pendency, or credit any overpayment, to Deposit Account No. 50-1214. Should no proper payment be enclosed herewith, as by a check being in the wrong amount, unsigned, post-dated, otherwise improper or informal or even entirely missing, the Commissioner is authorized to charge the unpaid amount to Deposit Account No. 50-1214.

John S. Paniaguas

John S. Paniaguas, Reg. No. 31,051

Katten Muchin Zavis Rosenman
Suite 1600
525 West Monroe Street
Chicago, Illinois 60661-3693
Telephone: (312) 902-5312
Facsimile: (312) 577-4532

PATENT

Attorney Docket N . 281450/208

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant(s): TEAC America, Inc.	**CERTIFICATE OF MAILING BY "EXPRESS MAIL"**
Serial No.: Herewith	"Express Mail" Mailing Label Number EV 162112455 US
Filing Date: October 9, 2003	Date of Deposit: October 9, 2003
Title: Complete Studio	I hereby certify that this paper or fee is being deposited with the United States Postal Service "Express Mail Post Office to Addressee" Service under 37 CFR §1.10 on the date indicated above and is addressed to the Commissioner for Patents, Washington, D.C. 20231.
Group Art Unit: N/A	
Examiner: N/A	Maria Flores (Typed or printed name of person mailing) Maria Flores (Signature of person mailing)

METHOD, APPARATUS, AND SYSTEM FOR SYNTHESIZING AN AUDIO PERFORMANCE USING CONVOLUTION AT MULTIPLE SAMPLE RATES (COMPLETE STUDIO)

BACKGROUND OF THE INVENTION

1. Field of the Invention

[0001] The present invention relates generally to acoustic processing and, more particularly, to a method, apparatus, and system for use in synthesizing an audio performance to acoustically vary one or more audible characteristics thereof using convolution at multiple sample rates.

2. Description of the Prior Art

[0002] Digital music synthesizers are known in the art. An example of a digital music synthesizer is U.S. Patent No. 5,502,747, which discloses based on multiple component filters and a hybrid time domain plus frequency domain processing. Unfortunately, the methodology utilized in the 5,502,747 patent is not efficient; and thus primarily only useful in academic and

scientific applications where computation time is not critical. Thus, there is a need for an efficient synthesizer that is more efficient than those in the prior art.

SUMMARY OF THE INVENTION

[0003] The present invention relates to a method, apparatus, and system for use in synthesizing an audio output which provides convolution – like audio processing at a greatly reduced processor load. The system is able to separate musical sources (instruments and other sound sources) form musical context; interactively recombine musical source and musical context with relatively accurate acoustical integrity, including surround sound contexts, microphone models, instrument resonance and digital audio effects overload on a musical source audio.

BRIEF DESCRIPTION OF THE DRAWING

[0004] FIGS. 1A and 1B are exemplary control panels for use with the present invention;

[0005] FIG. 2 is a high level block diagram of one embodiment of the present invention;

[0006] FIGS. 3A and 3B are block diagrams of an exemplary embodiment of a run-time input channel processing routine in accordance with the present invention;

[0007] FIG. 4 is a more detailed block diagram of the embodiment illustrated in FIG. 2;

[0008] FIG. 5 is a block diagram illustrating a process channel routine in accordance with the present invention;

[0009] FIG. 6 is a time domain representation of an exemplary audio input;

[0010] FIG. 7 is a block diagram of an audio collection and index sequencing system in accordance with the present invention;

[0011] FIG. 8 is a block diagram of a channel sequencing routine in accordance with the present invention; and

[0012] FIGS. 9 – 12 are more detailed block diagrams of the system.

DETAILED DESCRIPTION

[0013] FIGS. 1A and 1B illustrate graphical representations of exemplary embodiments of a control panel 100 which may be used in connection with the present invention. In the embodiment illustrated in FIG. 1A, the control panel 100 includes a drop-down menu 102 which may be used to select a predetermined musical context (*e.g.*, dark, hardwood floors, medium…), a drop-down menu 104 which may be used to select a "raw impulse", a drop-down menu 106 which may be used to select a particular musical instrument (*e.g.*, 1st violins, Legato down bows), a drop-down menu 108 which may be used to select an original microphone (*e.g.*, NT1000), and a drop-down menu 110 which may be used to select a particular replacement microphone (*e.g.*, AKG414). A display area 112 is provided for displaying a brief textual description of a microphone placement selection, as described in more detail below.

[0014] A button 114 is provided for selectively enabling and disabling a "cascade" feature associated with application of the raw impulse selected via the drop-down menu 104 to an audio

track. A button 116 is provided for selectively enabling and disabling an "encode" feature which permits the application of a user-selected acoustic model to the instrument selected via the drop-down menu 106. A display area 118 optionally may show a graphical or photographic representation of the musical context selected by the drop-down menu 102.

[0015] A button 120 is provided for selectively activating and deactivating a mid/side (M/S) microphone pair arrangement for left-side and right-side microphones. Additional buttons 121, 122, 123, and 124 are provided for specifying groups of microphones, including, for example, all microphones (button 121), front ("F") microphones (button 122), wide ("W") microphones (button 123), and rear or surround ("S") microphones (button 124).

[0016] The user also may enter microphone polar patterns and roll-off characteristics for each of the microphones employed in any given simulation. For that purpose, buttons 124, 125, 126, 127, 128, and 129 are provided for selecting a microphone roll-off characteristic or response. For example, buttons 125 and 126 select two different low-frequency bumps; button 127 selects a flat response, and buttons 128 and 129 select two different low-frequency roll-off responses, respectively. Similarly, buttons 130-134 allow a user to select one of several different well-recognized microphone polar patterns, such as an omni-directional pattern (button 130), a wide-angle cardioid pattern (button 131), a cardioid pattern (button 132), a hyper cardioid pattern (button 133), or a so-called "figure-8" pattern (button 134).

[0017] The control panel 100 also includes a placement control section 135, which, in the illustrated embodiment, contains a plurality of placement selector/indicator buttons (designated by numbers 1 through 18). These placement selector/indicator buttons allow a user to specify a

position of musical instruments within the user-selected musical context (*e.g.*, the position of the instrument selected by the drop-down menu 106 relative to the user-specified microphone(s)). The graphical display area 118 may display a depiction of the perspective of the room or musical context selected by the drop-down menu 102 corresponding to the placement within that room or musical context specified by the particular placement selector/indicator button actuated by the user. Of course, as will be readily apparent to those of ordinary skill in the art, many different alternative means may be employed to permit a user to select instrument placements within a particular musical context in addition to or instead of the placement selector/indicator buttons shown in **FIG. 1**. For example, a graphical depiction of the room or musical context could be displayed, and a mouse, trackball, or other conventional pointer control device could be used to move a location designator to a predetermined placement within the graphical depiction of the room or musical context corresponding to whatever placement within that room or musical context may be desired by the user.

[0018] As also shown in **FIG. 1**, the control panel 100 also includes a "mic-to-output" control section 136, which includes an array of buttons allowing a user to assign each microphone used in a given simulation to a corresponding mixer output channel. As shown, the control panel 100 provides for seven mixer output channels represented by the columns of buttons numbered one through seven in the mic-to-output control section 136. Seven mixer output channels allow for seven microphones to be used in a given simulation (*e.g.*, left and right front, left and right wide, left and right surround, and a center channel). Of course, those of ordinary skill in the art will readily appreciate that more or fewer mixer output channels may be provided in any given embodiment of the present invention based upon the needs of a particular simulator. For example, in a stereo simulator, only two mixer output channels need be provided. In order to

assign a particular microphone to a particular mixer output channel, the user need only depress the button in the row of buttons corresponding to the particular microphone and the column of buttons corresponding to the particular mixer output channel. The controls in each row of the mic-to-output control section 136 operate in a mutually exclusive fashion, such that a particular microphone can be associated only with one mixer output channel at a time.

[0019] The mic-to-output control section 136 also includes a button 140 for selectively enabling and disabling a "simulated stereo" mode in which a single microphone simulation or output is processed to develop two (*i.e.*, stereo) mixer output channels. This may be used, for example, to enable a simulated stereo output to be produced by a slow computer which does not have sufficient processing power to handle full stereo real-time processing. A button 142 is provided for selectively enabling a "true stereo" mode, which simply couples left and right stereo microphone simulations or outputs to two mixer output channels. Further, a button 144 is provided for selectively enabling and disabling a "seven-channel" mode in which each of seven microphone simulations or outputs is coupled to a respective mixer output channel to provide for full seven-channel surround sound output.

[0020] A button 146 is provided for selectively enabling and disabling a "tail extend" feature which causes the illustrated synthesizer to derive the first N seconds of the synthesized response by performing a full convolution and then to derive an approximation of the tail or terminal portion of the synthesized response using a recursive algorithm (described in more detail below) which is lossy but computationally efficient. Where exact acoustically simulation is not required, enabling the tail extend feature provides a trade-off between exact acoustical simulation and computational overhead. Associated with the tail extend feature are three

parameters, Overlap, Level, and Cutoff, and a respective slider control 148, 150, and 152 is provided for adjustment of each of these parameters.

[0021] More particularly, the slider control 148 permits adjustment of an amount of overlap between the recursively generated tail portion of the synthesized response or output signal and a time-wise prior portion of the output signal which is calculated by convolution at a particular sample rate. The slider control 150 permits adjustment of the level of the recursively generated portion of the output signal so that it more closely matches the level of the time-wise prior convolved portion of the output signal. The slider control 152 permits adjustment of the frequency-domain cutoff between the recursively generated portion of the output signal and the time-wise prior convolved portion thereof to thereby smooth the overall spectral damping of the synthesized response or output signal such that the frequency-domain bandwidth of the recursively generated portion of the output signal more closely matches the frequency domain bandwidth of the convolved portion thereof at the transition point between those two portions.

[0022] A plurality of further slider controls may be provided to allow a user to adjust the level corresponding to each microphone used in a particular simulation. In the illustrated embodiment, slider controls 154-160 are provided for adjusting recording levels of each of seven recording channels, each corresponding to one of the available microphones in the illustrated simulation or synthesizer system. In addition, a master slider control 161 is provided to allow a user to simultaneously adjust the levels set by each of the slider controls 154-160. As shown, a digital read-out is provided in tandem with each slider control 154-161 to indicate numerically to the user the level set at any given time by the corresponding slider control 154-161. In the illustrated embodiment, the levels are represented by 11-bit numbers ranging from 0 to 2047.

However, it should be evident to those of ordinary skill in the art that any other suitable range of levels in any suitable units could be used instead.

[0023] The control panel 100 also includes a level button 164, a perspective button 166, and a pre-delay button 168. The level button 164 allows a user to selectively activate and deactivate the level controls 154-161. The perspective button 166 allows the user to selectively activate and deactivate a perspective feature which allows the slider controls 154-161 to be used to adjust a parameter which simulates, for any given simulation, varying the physical dimensions of the musical context or room selected by the drop-down menu 102. The pre-delay button 168 allows the user to employ the slider controls 154-161 to adjust a parameter which simulates echo response speed (by adjusting the simulated lag between the initial echo in a recorded signal and a predetermined amount of echo density buildup).

[0024] Many modifications can be made in the graphical user interface shown and described herein to permit a user to adjust the various parameters of a simulation system in accordance with the principles of the present invention, and the graphical user interface shown and described herein is intended as exemplary only.

[0025] **FIG. 2** depicts a high-level block diagram of one exemplary embodiment of a simulation system 48 in accordance with the principles of the present invention. The simulation system 48 includes a runtime input channel processing routine 50, a runtime sequencing, control, and data manager 52, a process-channel module 53 containing a multi-rate adaptive filter 54, a collection and alignment routine 56, and a tail extension processor 58. As shown, input digital music samples are provided to the runtime input channel processing routine 50, which processes those

samples and provides the processed samples to the runtime sequencing, control and data manager 52. In addition, impulse response data representing, for example, the frequency-domain impulse responses corresponding to the characteristics of user-selected microphones, musical context musical instruments, and relative positioning of the user selected microphones and/or musical instruments within the user selected musical context are stored in a coefficient storage 60. A loadtime coefficient processing routine 62 and a runtime coefficient processing routine 64 successively process coefficients from the coefficient storage 60 based on the user input 66 provided via, for example, a control panel or graphical user interface such as that depicted in **FIG. 1**. The runtime sequencing, control, and data manager 52 provides processed musical input samples and processed impulse response coefficients to the above-described process channel module 53 which produces audio output samples 68 based on the processed audio input samples and processed impulse response coefficients in accordance with the principles of the present invention.

[0026] **FIG. 3** illustrates a block diagram of one exemplary embodiment of the runtime input channel processing routine 50 shown in **FIG. 2**. As shown in **FIG. 3,** the runtime input channel processing routine 50 receives digital sample data from a digital sample buffer 70 and produces frequency domain output data, at least some of which is passed to a frequency domain buffer 94. Some of the frequency domain output data is also passed to a frequency domain buffer 96. In accordance with the principles of the present invention, this data may be provided at a sample rate that is lower than the sample rate of the frequency domain data passed to the frequency domain buffer 94 as explained in more detail below.

[0027] Digital samples from the digital sample buffer 70 are copied on a frame-by-frame basis by frame copy routines 72 and 74 to respective frame buffers 76 and 78. More particularly, samples from a time-domain initial portion (B) of an audio input sample are copied by the frame copy routine 72 into a frame buffer 76, and digital samples from a time-domain subsequent portion (A) of the audio input sample are copied by the frame copy routine 74 to the frame buffer 78. A downsampling tail maintenance routine 80 is coupled between the digital sample buffer 70 and the frame buffer 78. A fast fourier transform (FFT) module 82, including FFT routines 84, 86, and 88, is provided for converting frames of data, which are represented in the time domain in frame buffers 76 and 78 into corresponding frequency-domain data. More particularly, the FFT routine 84 produces a fast fourier transform of a frame from the frame buffer 76 and provides the transformed data to a frequency domain buffer 94. Frame data from the frame buffer 78 is filtered by a low-pass filter (a 2:1 filter in the illustrated embodiment) and is then passed through a decimation frame buffer 92 to the FFT routine 86 which performs an FFT on the decimated, filtered frame data and stores the resulting frequency domain frame data in a frequency domain buffer 96.

[0028] In the event a user wishes not to employ downsampled convolution (*i.e.*, preferring instead to achieve the acoustic accuracy of full-sample-rate convolution by expending greater processing power), the FFT module 88 may transform the frame data from the frame buffer 78 at its original sample rate and thus provide full-sample-rate frequency domain data to the frequency domain buffer 96. Operation of the frame copy routines 72 and 74, the downsampling tail maintenance routine 80, the FFT module 82, and the low-pass filter 90 is handled by a frame control process 98. Further detail regarding the operation of the run-time input channel

processing routine 50 and the components thereof as described herein is presented in the source code included in the Appendix hereto.

FIG 4 depicts a block diagram illustrating in greater detail the simulation system shown in **FIG 2**, including an expanded illustration of the flow of data that occurs in operation of that system. As show, a run time memory 100 includes a plurality of frame buffers 102, 103, 104 for storing audio input samples received for example, N audio input channels. Each of the frame buffers 102, 103, 104 is sized to received one frame of input audio samples at a time from a corresponding one of the N audio input channels. The run time memory 100 also includes a plurality of data structures 106, 107, 108 for storing coefficients representing M impulse responses. Impulse response data may be retrieved from a co-efficient mass storage 109 by a load and process routine 110 in response to a user command supplied to the load and process routine 110 via an IO control module 111. The run time memory 100 also includes a plurality of T output buffers 112, 113, 114, each of which is sized to receive one frame of output audio samples at a time for communication to a respective one of T output sample streams. The run time memory 100 also includes an instance data storage 116 for storing a plurality of J instances.

As shown in **FIG 4**, data from the frame buffers 102, 103, 104 and the data structures 106, 107, 108 is communicated to a channel sequencing module 118 which serves to time-multiplex the data for processing by a process channel - channel module 120. In particular, information passed from the channel sequencing module 118 to the process channel module 120 includes, for each of the N audio input channels, data representing a time wise first portion of each frame of data received via that audio input channel (PXLBf(i), i = 1, 2,...N), data representing a time-wise second portion of a frame of data received via that audio input channel (PXLAf(i), i = 1,2, ...N),

PIc(i), PIOBuf(i), dwFRAMESize, PI, simulated stereo, M/S decode, and control. All of this data passes from the channel sequencing module 118 to the process channel routine 120. As also shown in **FIG 4**, by directional communication as provided between the process channel routine 120 and the run time memory 100, indicated by arrows 122.

FIG 5 depicts a block diagram illustrating in greater detail the operation of the process channel routine 68 (**FIG 2**) as described above. As shown in **FIG 5**, dynamic channeled data 150, which may be stored in the above-described run time memory 100 (**FIG 4**), may comprise a data structure including a field 152 which contains an index reference each index B into a second portion of the impulse response for the channel under consideration. The data structure described herein is used for data pertaining to audio input samples received on one audio input channel. Those skilled in the art will appreciate that other instances of this data structure also will be used in connection with processing data from other of the audio input channels in accordance with the principles of the present invention. The data structure 150 includes at least fields 152, 154, 156, 158, 160, 162, 164, 166, and 168. The field 152 contains (1) an index reference (h index B) into a time wise first portion of impulse response data for that audio input channel to indicate a current position within that data where a calculation is being performed; (2) an index reference (h index A) into a time wise second portion of the impulse response data for that channel to indicate a position within that second portion where calculation is being performed; and (3) additional control data.

For purposes of illustration, a time-domain representation of an exemplary audio input signal is shown graphically in **FIG 6**. As shown in **FIG 6**, the illustrated audio input signal includes a time wise first portion designated B) and a continuous, time wise second portion designated A, and a "tail" portion that extends continuously beyond the time wise second portion A. In the

time domain, the audio input signal may be partitioned into groups of samples, wherein the size of the groups of samples is denoted XLENAII (the major frame size for FFT blocks) as indicted in **FIG 6**. The first portion of the audio input signal (herein after referred to as the "B portion") preferably includes an number of samples corresponding to the major frame size for FFT blocks XLENAII, and the time wise second portion (herein after referred to as the "A portion") preferably is made up of a number of such frames of samples the total number of samples making up the audio signal illustrated in **FIG 6** is denoted by FTAPSII. A pointer hindex is used to designate a relative position within the aggregate collection of samples making up the illustrated impulse response or audio input signal.

The field 154 (**FIG 5**) contains a pointer to a data structure populated with a frequency domain representation of an acoustic model being simulated (*e.g.*, a particular acoustic space, a particular microphone, a particular musical instrument body residence characteristic, etc.). This filter impulse response (FIR) that is stored in the data structure Hx is sized to accommodate twice the number of samples making up the acoustic model (*i.e.*, IMPSIZE*2).

The field 156 contains an intermediate part of the product of a vector multiplication of the frequency domain impulse response Hx and the impulse response. The field 158 receives a time-domain equivalent of the frequency domain impulse response portion stored in the field 156.

The field 160 contains indices to past and present frames in the audio collection buffer for the B portion of the impulse response (acolindexprevB and acolindexB, respectively, and for the A portion of the impulse response, acolindexprevA and acolindexB, respectively.

The field 162 contains a pointer to the audio collection buffer (and intermediate accumulative that facilitates the overlap-add) and which comprehends frame-based overlap and modulo

addressing. This buffer acol, is modulo addressed and is sized of length impulse size (the time-domain length of the impulse response) into which successive frames are overlap added.

A table identifying variable names is provided below.

This data structure is fed into Process Channel by means of single pointer to an instance of this data structure. The Channel Sequencing chooses the particular pointers and controls that are fed into process channel. Each instance of this data structre represents one dynamic channel data impulse block in Runtime Memory

Piecewise Convolution is done (in portions which are combined) by an "overlap-add" method (technically, overlap-subtract with a downstream phase reversal)

typedef struct _tagDynamicChannelData
{

Data Type	Variable/Field	Description
Ipp32f 32-bit flo point	Hx[IMPSIZE*2];	//filter impulse response (FIR Filter), (Frq domain representation of the acoustic model being simulated (e.g., acoustic space, microphone, musical instrument body resonance))
Ipp32f	acol[ACOLLENGTH];	//audio collect (intermediate accumulator for overlap-add)–comprehends frame-based overlap and modulo addressing modulo addressed buffer sized of length impulsesize (the time-domain length of the impulse response) into which successive frames are overlap added
Ipp32f	acolH[(ACOLLENGTH/2) + LPFTAPS];	//turbo collect, Half size for reduced sample-rate to save CPU overhead
Ipp32f	acolDH[(ACOLLENGTH/2) + LPFTAPS];	//turbo tail ext Delay collect, Half size for reduced sample-rate to save CPU overhead
Ipp32f	acolD[ACOLLENGTH];	//tail extension delay
Ipp32f	tarMicLevel;	//target mic level scale (new setpoint)
Ipp32f	delMicLevel;	//delta mic level scale (recursively calculated transition level gradulated from one sample to the next to smooth the transition)
Ipp32f	runMicLevel;	//runtime mic level scale (original being applied)
Ipp32f	runMicPerspec;	//runtime mic perspective scale (affects the envelope of the impulse response–used for runtime scaling) applies a volume scaling to early part of response that increases or decreases volume to create perception of a close or distant perspective on the audio–correlated to some scape value that makes sense for the user interface of the apparent distance of the sound source from the mic.
Ipp32f	tarDirectLevel;	//target Direct Sim stereo level scale
Ipp32f	delDirectLevel;	//delta Direct Sim stereo level scale
Ipp32f	runDirectLevel;	//runtime Direct Sim stereo level scale
Ipp32f	alignDirect;	//alignment dummy
DWORD unsigned i	hindexA;	//sub frame frq H[hindex] - index reference into impulse response (the current potion on which a calculation is being performed (A portion, which may be at a lower sample rate)
DWORD	acolindexA;	//audio collect buffer index - current index into collect buffers used for overlap-add
DWORD	acolindexprevA;	//previous frame collection buffer index - previous index value in the collection or accumulation
DWORD	outindex;	//collect buffer output index - index to where data can be read from audio collect buffer (second from top above) and sent to output buffer
DWORD	hindexB;	//sub frame frq H[hindex]
DWORD	acolindexB;	//audio collect buffer index
DWORD	acolindexprevB;	//previous frame collection buffer index
DWORD	dummyxxxx;	//alignment dummy (placeholder)
DWORD	dlyWindex;	//tail extension delay Write index (delay needed to align modeled portion of the impulse response with the actual result of the impulse response)
DWORD	dlyRindex;	//tail extension delay Read index (delay needed to align modeled portion of the impulse response with the actual result of the impulse response)
DWORD	te1A;	//tail extension state variable (for Tail Extension Filter)
DWORD	te2A;	//tail extension state variable

DWORD	FcFbk;	//tail extension state variable
DWORD	FcFbk2;	//tail extension state variable
DWORD	FcFbk3;	//tail extension state variable
DWORD	FcFbk4;	//tail extension state variable
DWORD	FcFbk5;	//tail extension state variable
DWORD	FcFbk6;	//tail extension state variable
DWORD	aligntel;	//16 byte alignment dummy
DWORD	aligntе2;	//16 byte alignment dummy
Ipp32f	AP1[ALL_PASS_SAMPLES];	//tail extension, all-pass buffer
Ipp32f	AP2[ALL_PASS_SAMPLES];	//tail extension, all-pass buffer
Ipp32f	SSt[SIM_STEREO_SAMPLES];	//sim stereo delay buffer (for slow processors; simulates stereo audio by using stereo audio filters and complementary comb filters. OPTIONAL
Ipp32f	SStDD[SIM_STEREO_SAMPLES];	//sim stereo, direct delay buffer
DWORD	AP1_r;	//AP buffer read index
DWORD	AP2_r;	//AP buffer read index
DWORD	SSt_r;	//Sim Stereo buffer read index
DWORD	AP1_w;	//AP buffer write index offset
DWORD	AP2_w;	//AP buffer write index offset
DWORD	tarSSt_w;	//target Sim Stereo buffer write index offset
Ipp32f	tarSStWidth;	//target Sim stereo depth
Ipp32f	delSStWidth;	//delta Sim stereo depth
Ipp32f	runSStWidth;	//runtime Sim stereo depth
DWORD	delSSt_w;	//delta Sim Stereo buffer write index offset
DWORD	runSSt_w;	//runtime Sim Stereo buffer write index offset
DWORD	SStDD_w;	//sim stereo, direct delay write offset

```
} DYNAMICCHANNELDATA, *PDYNAMICCHANNELDATA;

#endif
```

FIGS. 7 – 12 contain additional details of the invention. A complete copy of the source code is included in a CD appendix.

The foregoing description is for the purpose of teaching those skilled in the art the best mode of carrying out the invention and is to be construed as illustrative only. Numerous modifications and alternative embodiments of the invention will be apparent to those skilled in the art in view of this description, and the details of the disclosed structure may be varied substantially without departing from the spirit of the invention. Accordingly, the exclusive use of all modifications within the scope of the appended claims is reserved.

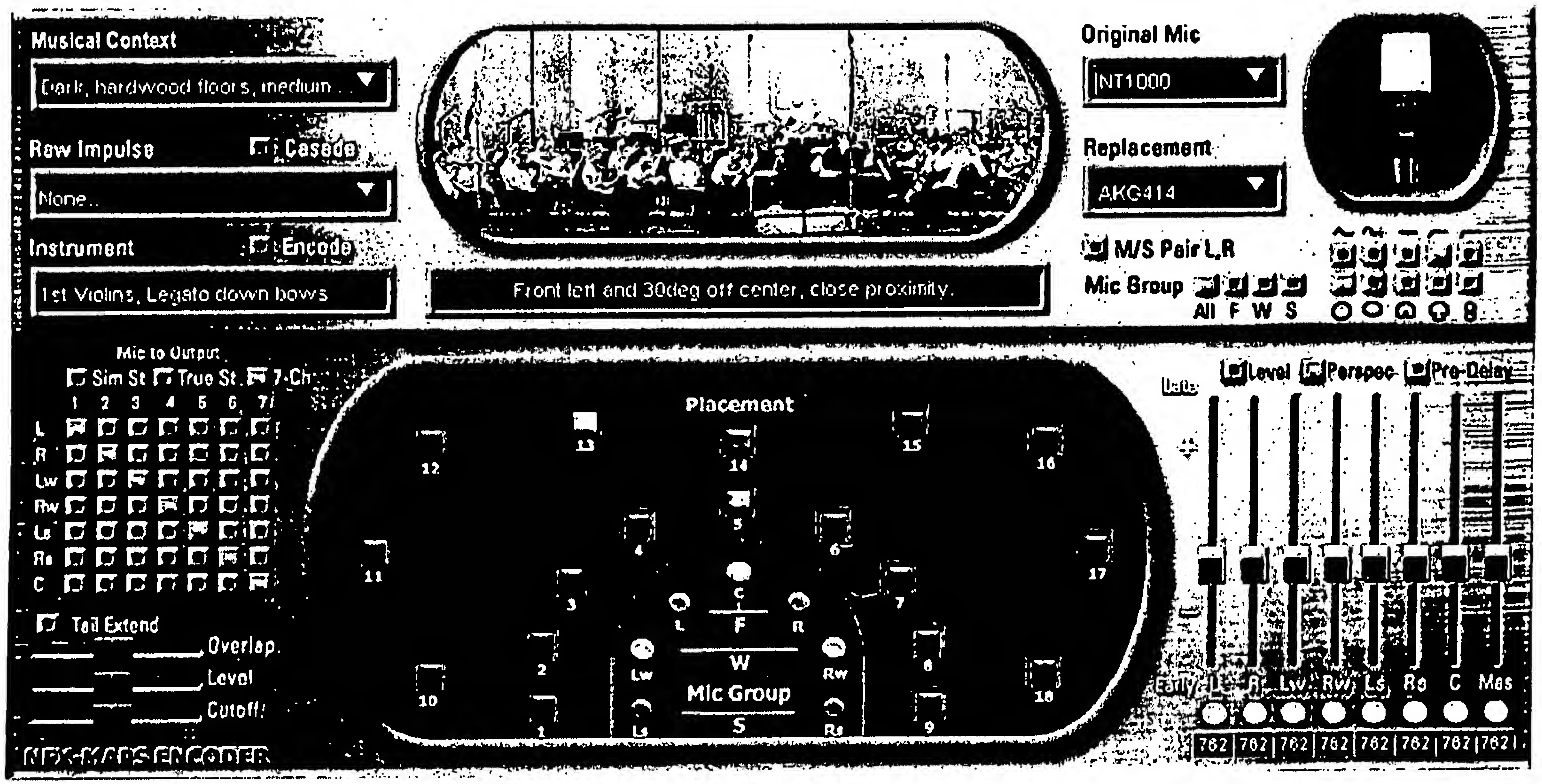

Musical Context
Dark, hardwood floors, medium ...
Raw Impulse
Cascade
None..
Instrument
Encode
1st Violins, Legato down bows
Front left and 30deg off center, close proximity.
Original Mic
NT1000
Replacement
AKG414
M/S Pair L,R
Mic Group
All F W S
Mic to Output
Sim St
True St.
7-Ch
1 2 3 4 5 6 7
L
R
Lw
Rw
Ls
Rs
C
Tail Extend
Overlap
Level
Cutoff
NFX-MAPS ENCODER
Placement
12 13 14 15 16
11 4 5 6 17
3 7
C
L F R
2 8
Lw W Rw
10 Mic Group 18
1 Ls S Rs 9
Level
Perspec
Pre-Delay
Early L R Lw Rw Ls Rs C Mas
762 762 762 762 762 762 762 762

FIG. 1

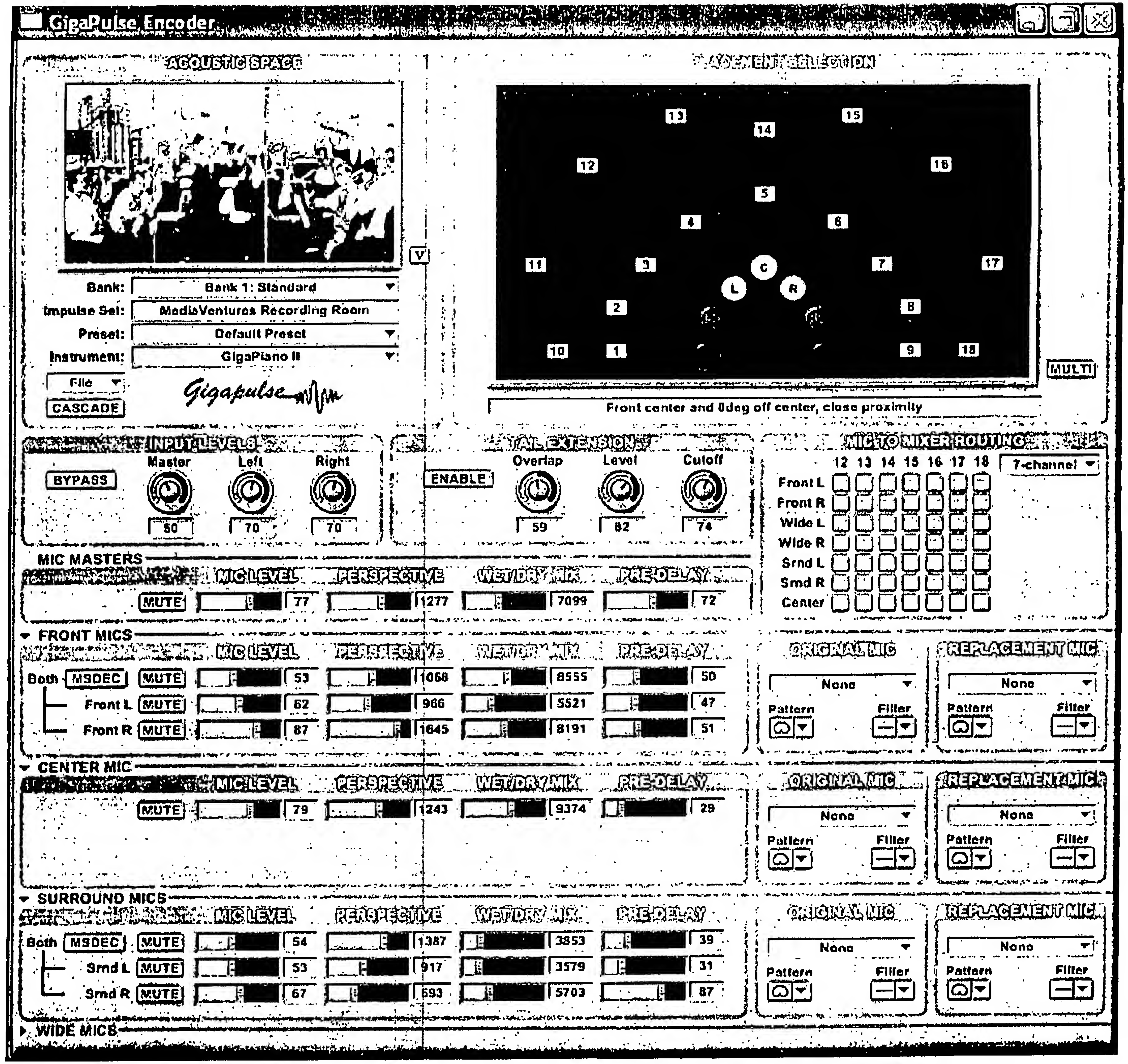

GigaPulse Encoder
ACOUSTIC SPACE
PLACEMENT SELECTION
Bank: Bank 1: Standard
Impulse Sel: MediaVentures Recording Room
Preset: Default Preset
Instrument: GigaPiano II
File
CASCADE
Gigapulse
MULTI
Front center and 0deg off center, close proximity
INPUT LEVELS
BYPASS
Master Left Right
50 70 70
TAIL EXTENSION
ENABLE
Overlap Level Cutoff
59 82 74
MIC TO MIXER ROUTING
12 13 14 15 16 17 18
7-channel
Front L
Front R
Wide L
Wide R
Srnd L
Srnd R
Center
MIC MASTERS
MIC LEVEL PERSPECTIVE WET/DRY MIX PRE-DELAY
MUTE 77 1277 7099 72
FRONT MICS
Both MSDEC MUTE 53 1068 8555 50
Front L MUTE 62 966 5521 47
Front R MUTE 87 1645 8191 51
ORIGINAL MIC
REPLACEMENT MIC
None
Pattern
Filter
CENTER MIC
MUTE 79 1243 9374 29
SURROUND MICS
Both MSDEC MUTE 54 1387 3853 39
Srnd L MUTE 53 917 3579 31
Srnd R MUTE 67 693 5703 87
WIDE MICS

FIG 1B

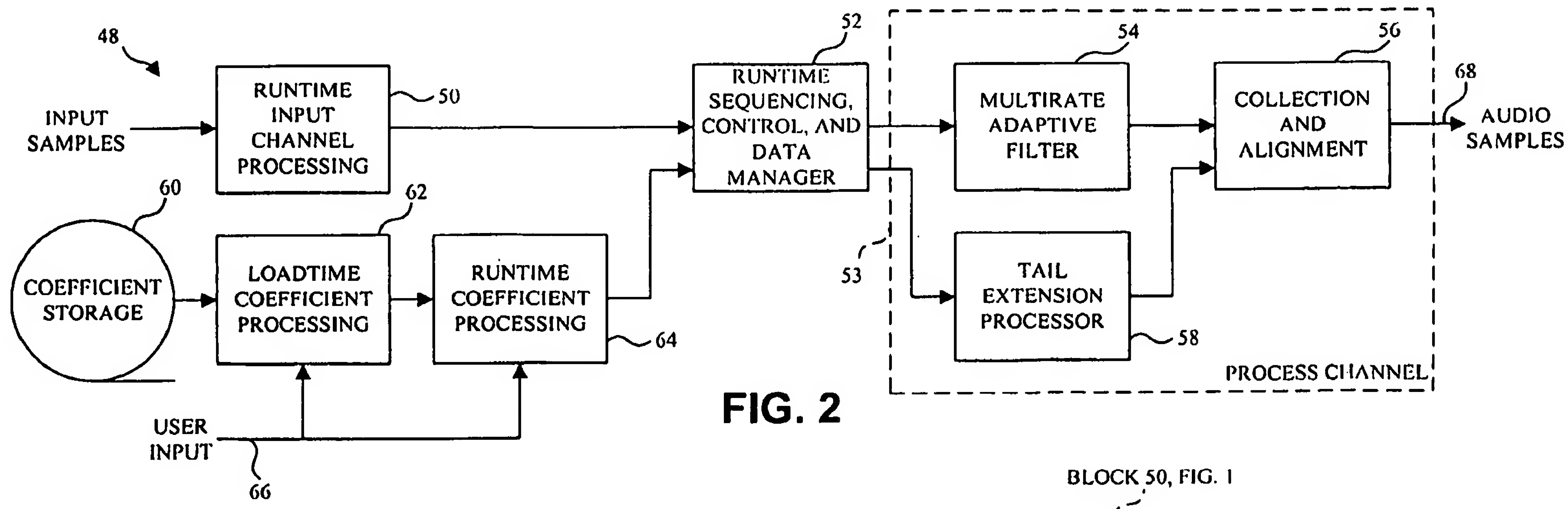

48
INPUT SAMPLES
RUNTIME INPUT CHANNEL PROCESSING
50
COEFFICIENT STORAGE
60
LOADTIME COEFFICIENT PROCESSING
62
RUNTIME COEFFICIENT PROCESSING
64
USER INPUT
66
RUNTIME SEQUENCING, CONTROL, AND DATA MANAGER
52
53
MULTIRATE ADAPTIVE FILTER
54
TAIL EXTENSION PROCESSOR
58
COLLECTION AND ALIGNMENT
56
PROCESS CHANNEL
68
AUDIO SAMPLES

FIG. 2

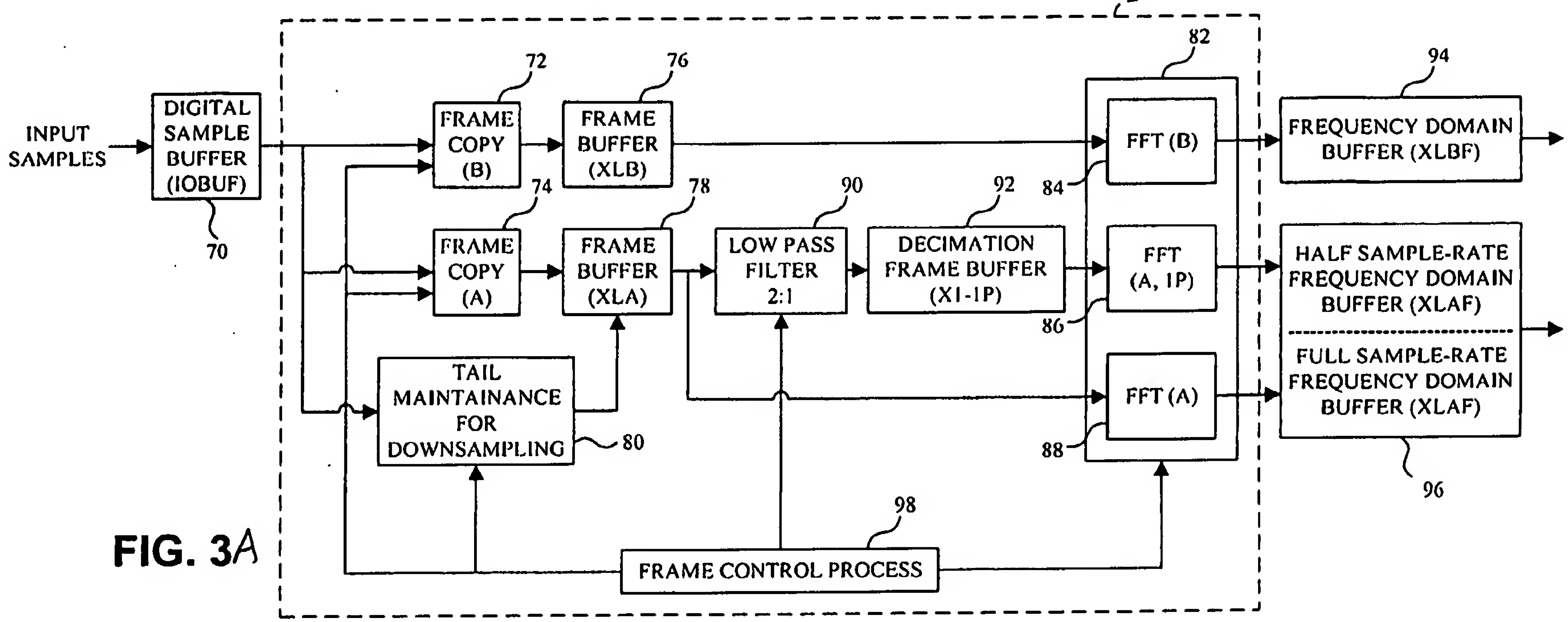
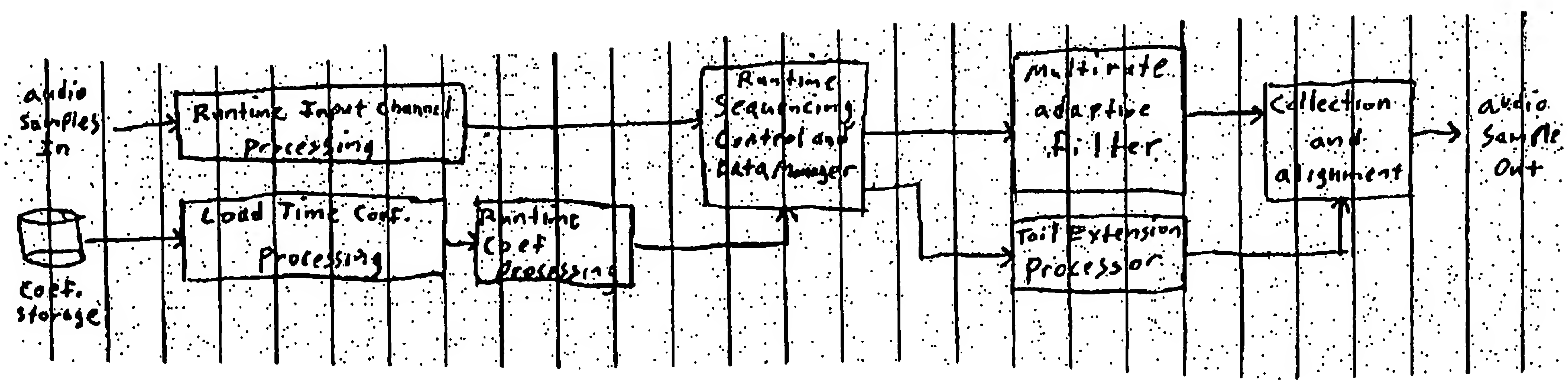

BLOCK 50, FIG. 1
INPUT SAMPLES
DIGITAL SAMPLE BUFFER (IOBUF)
70
FRAME COPY (B)
72
FRAME BUFFER (XLB)
76
FRAME COPY (A)
74
FRAME BUFFER (XLA)
78
LOW PASS FILTER 2:1
90
DECIMATION FRAME BUFFER (X1-1P)
92
TAIL MAINTAINANCE FOR DOWNSAMPLING
80
FRAME CONTROL PROCESS
98
82
FFT (B)
84
FFT (A, 1P)
86
FFT (A)
88
FREQUENCY DOMAIN BUFFER (XLBF)
94
HALF SAMPLE-RATE FREQUENCY DOMAIN BUFFER (XLAF)
FULL SAMPLE-RATE FREQUENCY DOMAIN BUFFER (XLAF)
96

FIG. 3A



audio Samples In
Runtime Input Channel Processing
Coef. Storage
Load Time Coef. Processing
Runtime Coef Processing
Runtime Sequencing Control and Data Manager
Multirate adaptive filter
Tail Extension Processor
Collection and alignment
audio Sample Out

FIG 3B

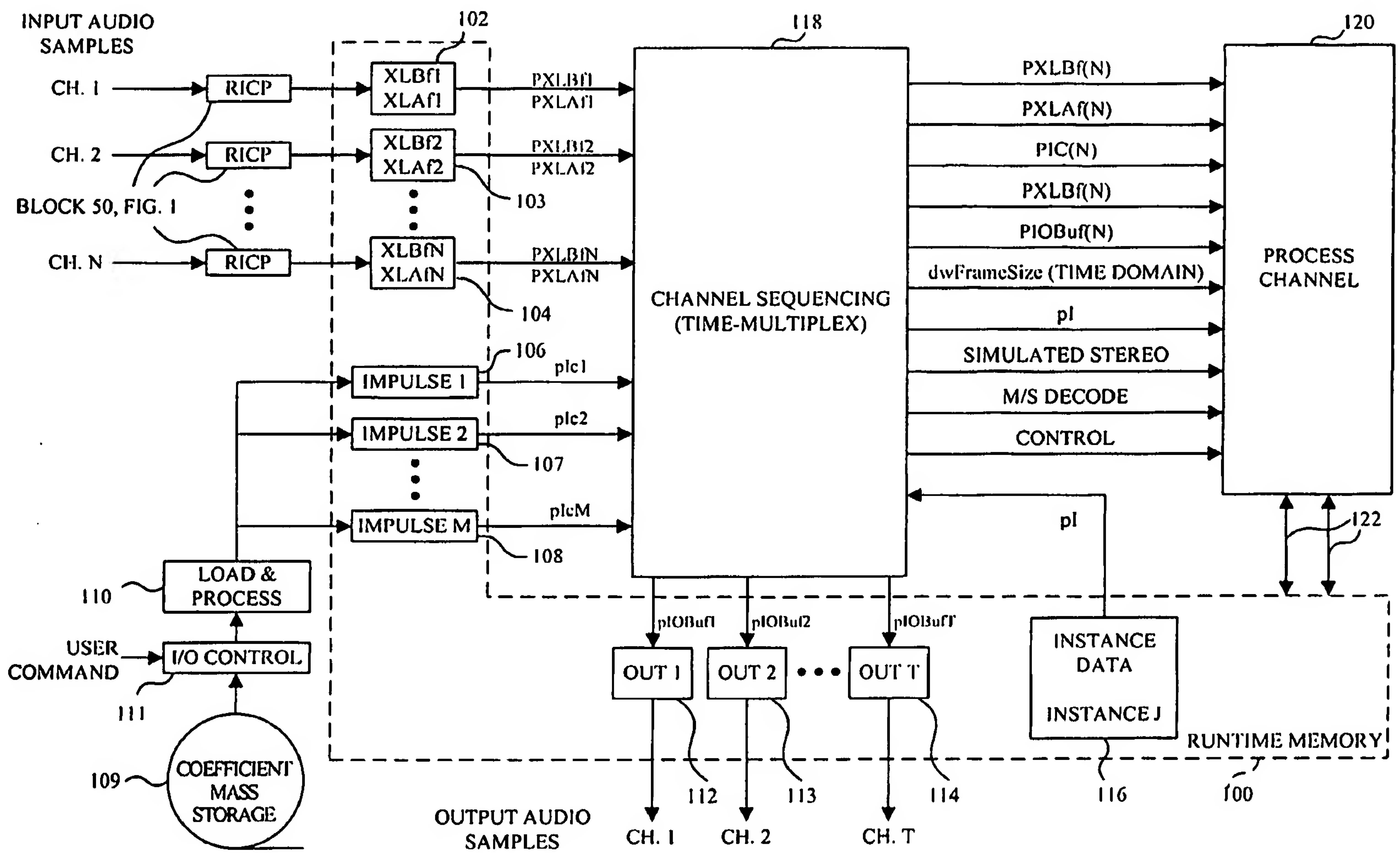

INPUT AUDIO SAMPLES
CH. 1
CH. 2
CH. N
BLOCK 50, FIG. 1
RICP
RICP
RICP
102
XLBf1 XLAf1
XLBf2 XLAf2
XLBfN XLAfN
103
104
PXLBf1 PXLAf1
PXLBf2 PXLAf2
PXLBfN PXLAfN
IMPULSE 1
IMPULSE 2
IMPULSE M
106
107
108
pIc1
pIc2
pIcM
LOAD & PROCESS
110
USER COMMAND
I/O CONTROL
111
COEFFICIENT MASS STORAGE
109
118
CHANNEL SEQUENCING (TIME-MULTIPLEX)
PXLBf(N)
PXLAf(N)
PIC(N)
PXLBf(N)
PIOBuf(N)
dwFrameSize (TIME DOMAIN)
pI
SIMULATED STEREO
M/S DECODE
CONTROL
120
PROCESS CHANNEL
122
pI
pIOBuf1
pIOBuf2
pIOBufT
OUT 1
OUT 2
OUT T
112
113
114
INSTANCE DATA
INSTANCE J
116
RUNTIME MEMORY
100
OUTPUT AUDIO SAMPLES
CH. 1
CH. 2
CH. T

FIG. 4

Process Channel (revised 7-18-2003)

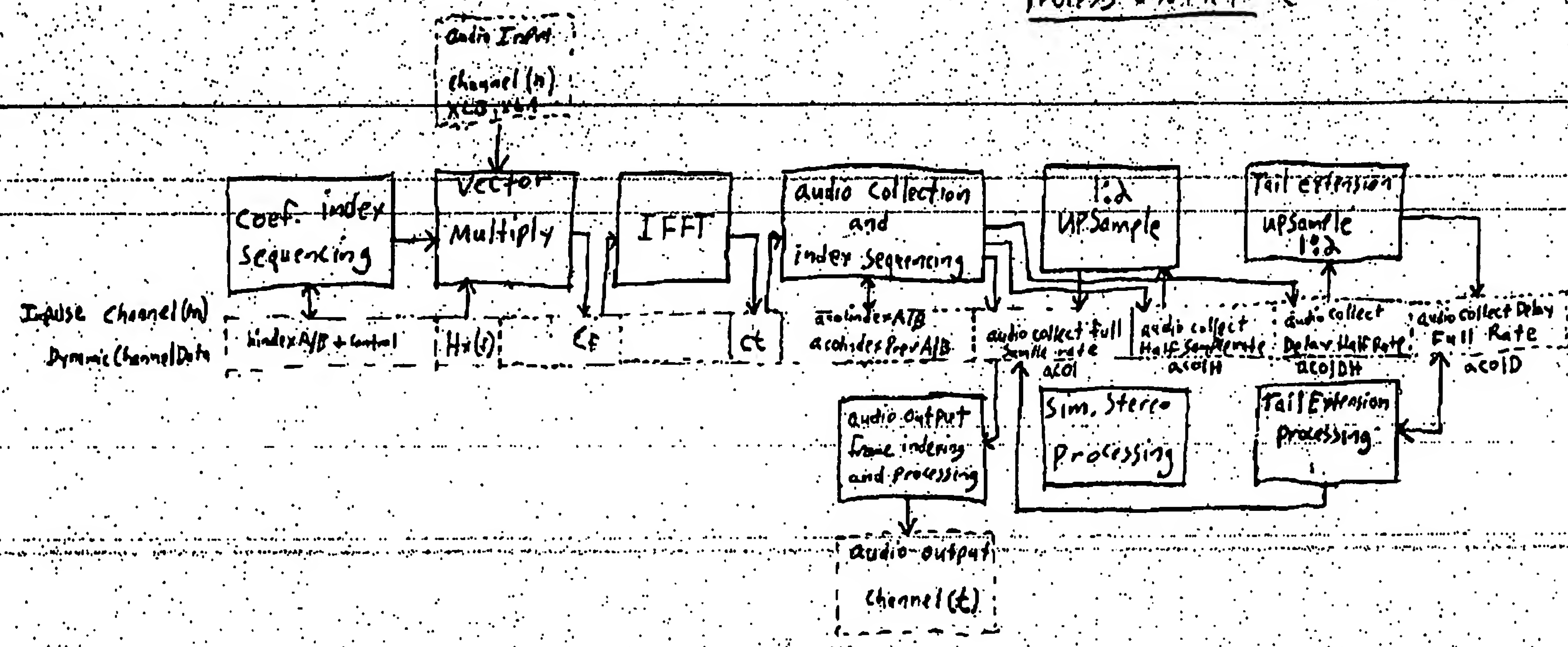


FIG 5

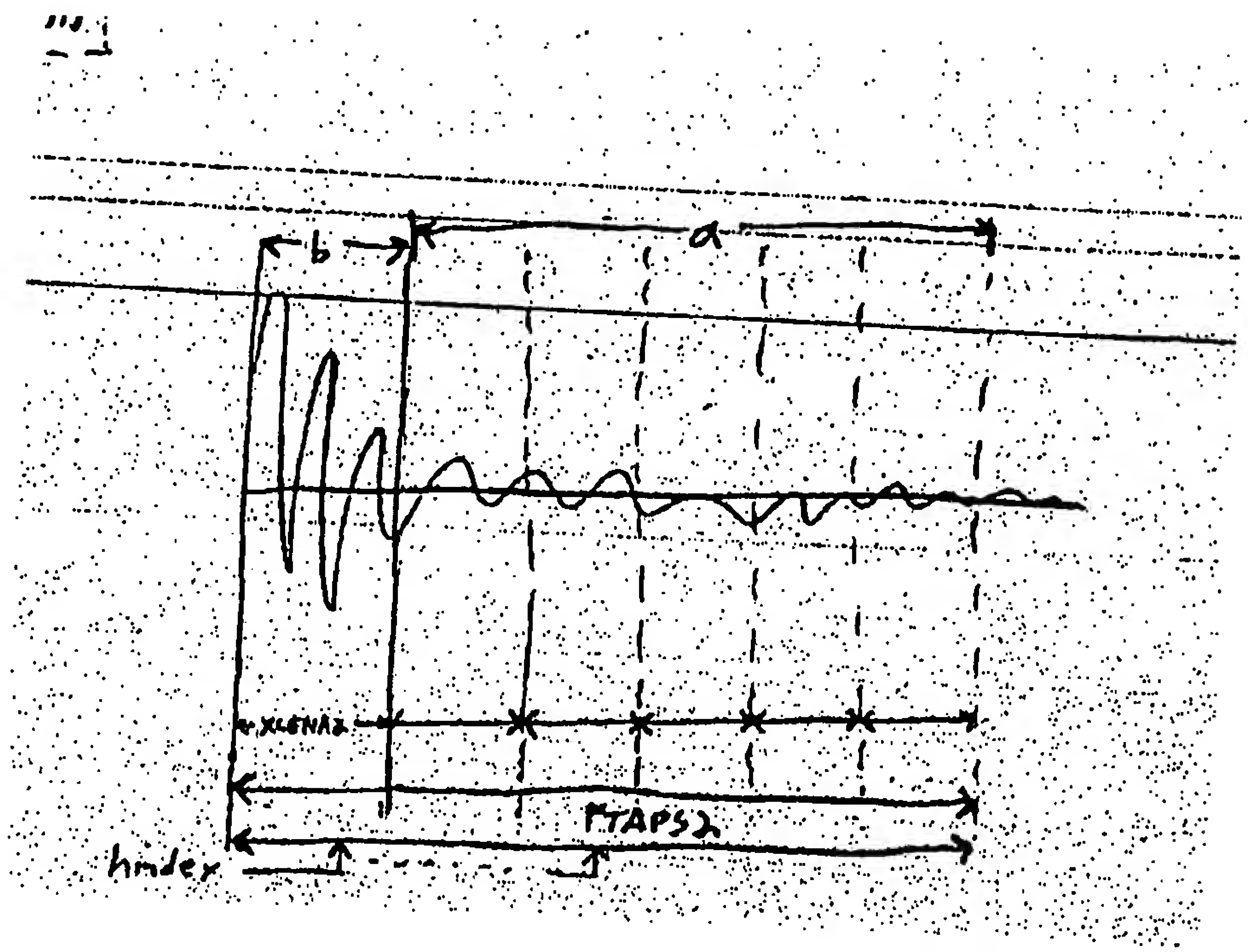


FIG 6

AUDIO COLLECTION & INDEX SEQUENCING

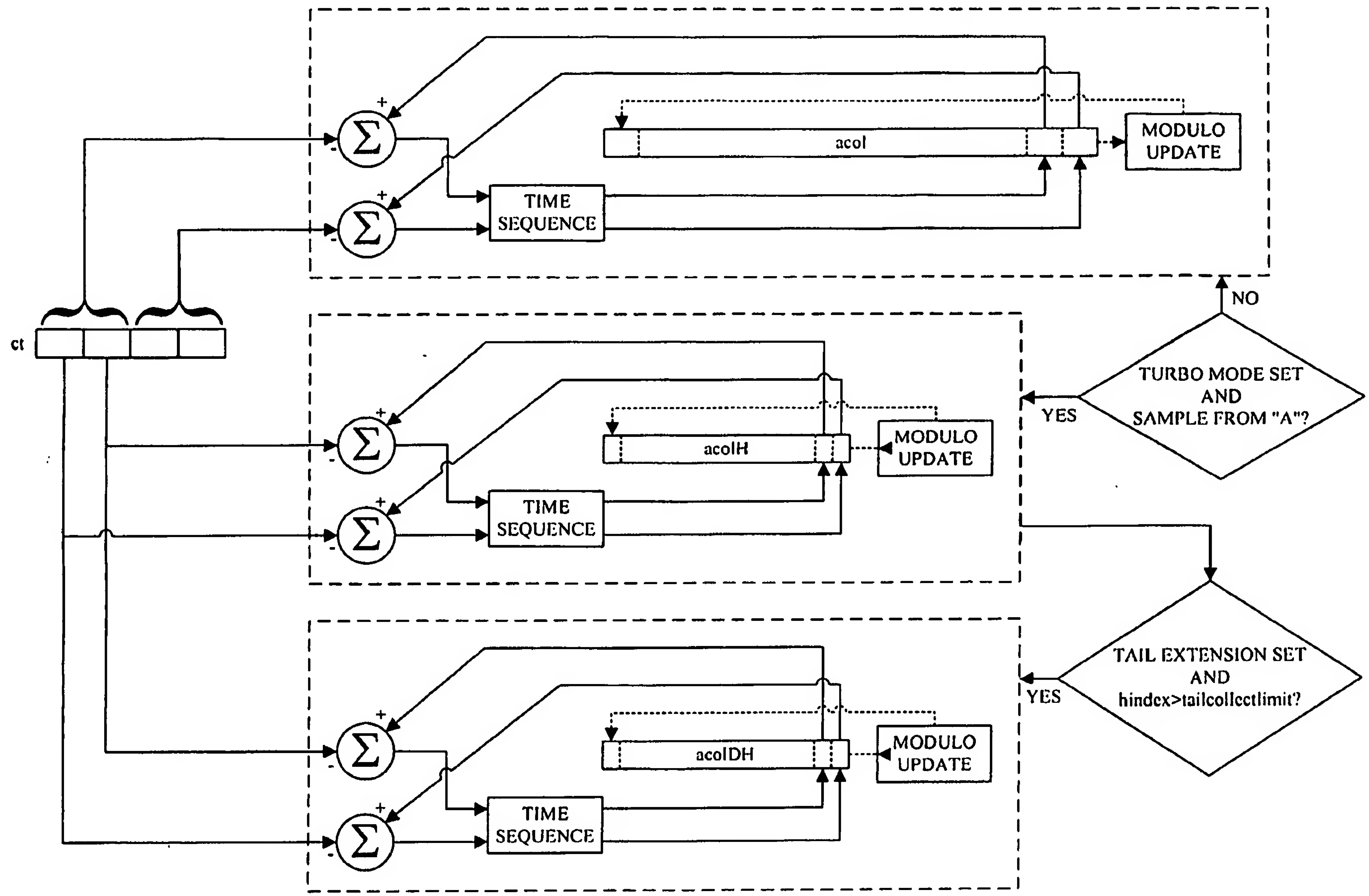
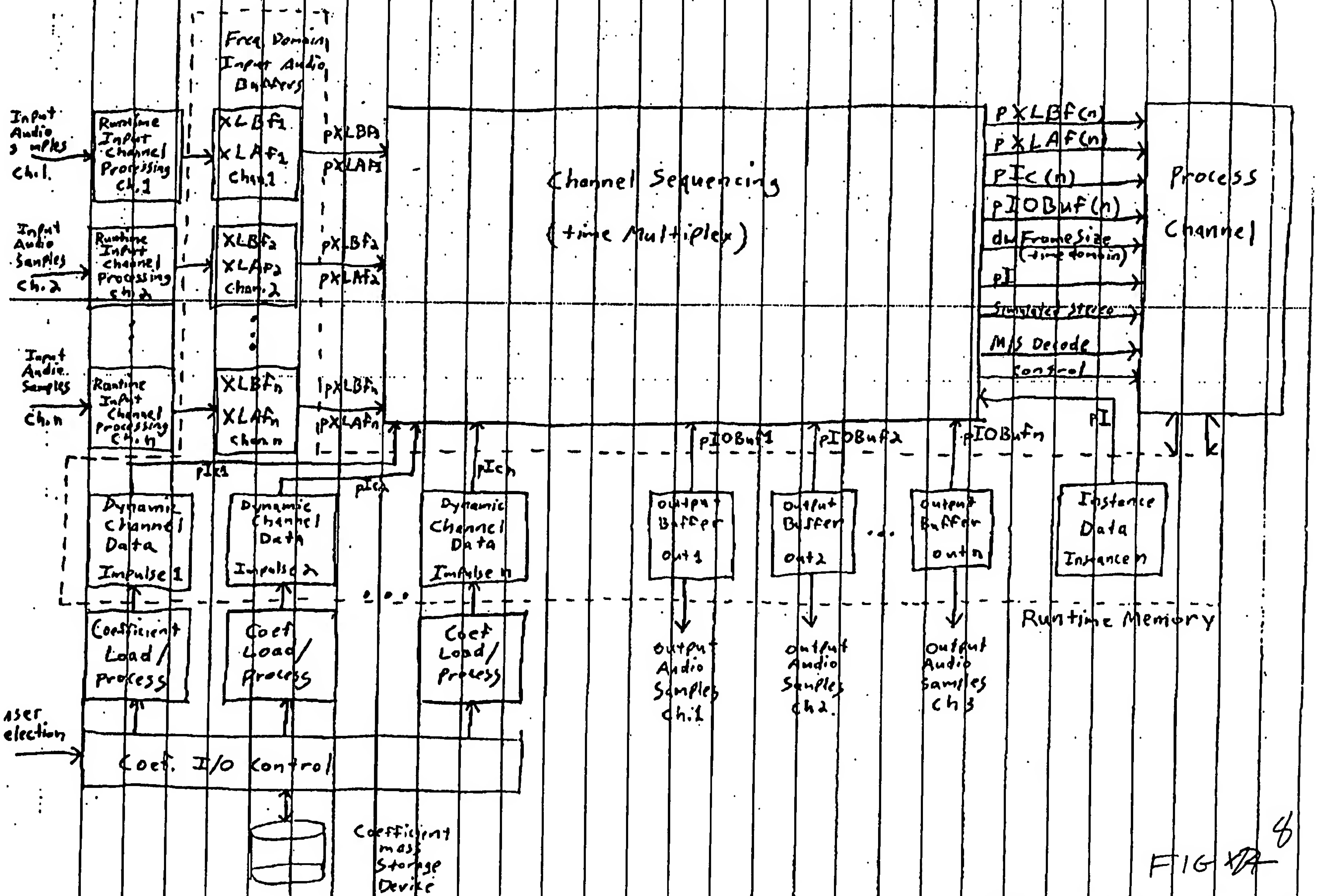


FIG. 7



Input Audio Samples Ch.1
Input Audio Samples Ch.2
Input Audio Samples Ch.n
Runtime Input Channel Processing Ch.1
Runtime Input Channel Processing Ch.2
Runtime Input Channel Processing Ch.n
Freq Domain Input Audio Buffers
XLBf1 XLAf1 Chan.1
XLBf2 XLAf2 Chan.2
XLBfn XLAfn Chan n
pXLBf1
pXLAf1
pXLBf2
pXLAf2
pXLBfn
pXLAfn
Channel Sequencing
(time Multiplex)
pXLBf(n)
pXLAf(n)
pIc(n)
pIOBuf(n)
dwFrameSize (time domain)
pI
Simulated Stereo
M/S Decode
Control
Process Channel
pIc1
pIc2
pIcn
pIOBuf1
pIOBuf2
pIOBufn
pI
Dynamic Channel Data Impulse 1
Dynamic Channel Data Impulse 2
Dynamic Channel Data Impulse n
Output Buffer Out 1
Output Buffer Out 2
Output Buffer Out n
Instance Data Instance n
Runtime Memory
Output Audio Samples Ch.1
Output Audio Samples Ch.2
Output Audio Samples Ch 3
Coefficient Load/Process
Coef Load/Process
Coef Load/Process
User Selection
Coef. I/O Control
Coefficient mass Storage Device

FIG 8

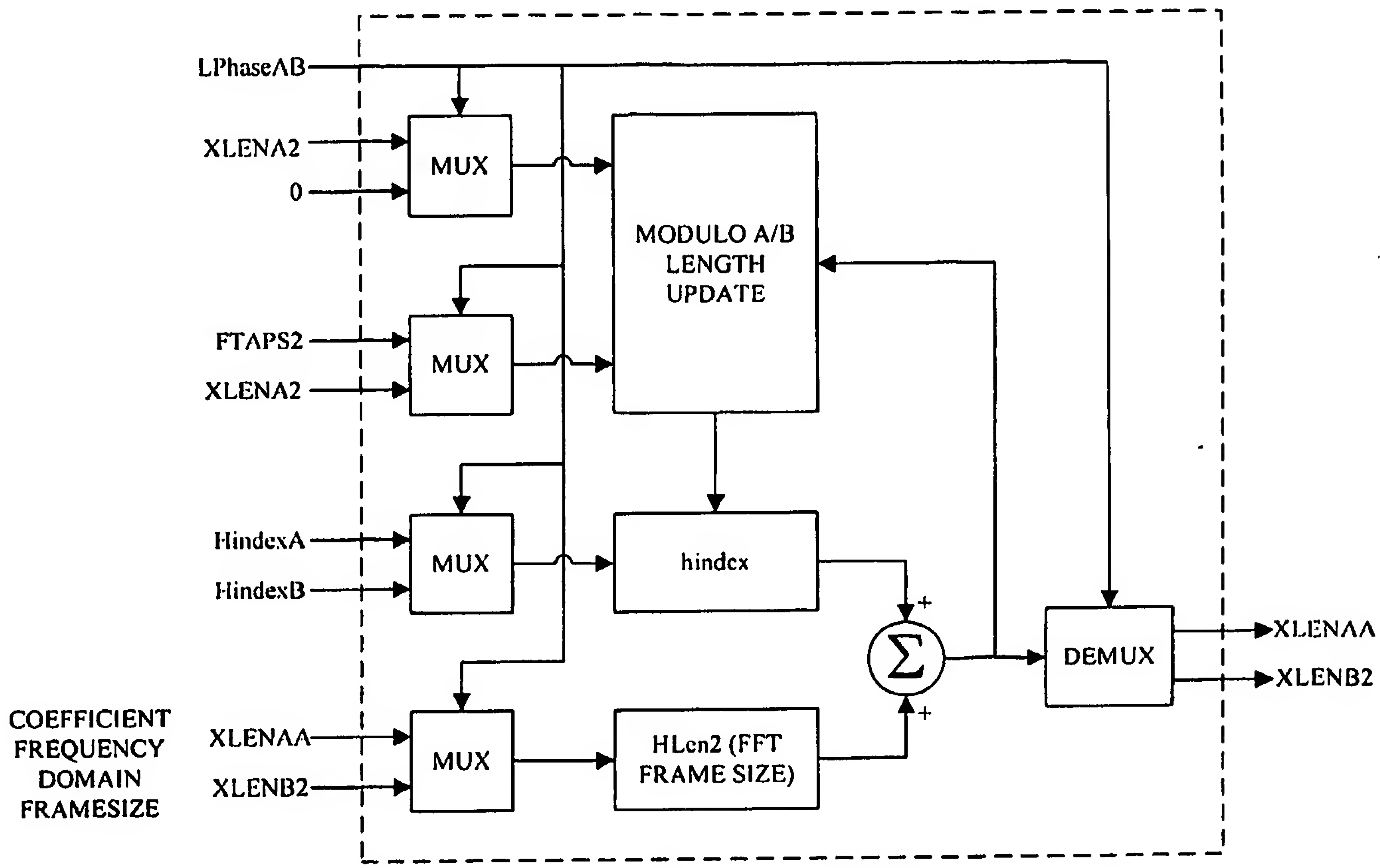

LPhaseAB
XLENA2
0
MUX
FTAPS2
XLENA2
MUX
HindexA
HindexB
MUX
COEFFICIENT FREQUENCY DOMAIN FRAMESIZE
XLENAA
XLENB2
MUX
MODULO A/B LENGTH UPDATE
hindex
HLen2 (FFT FRAME SIZE)
Σ
+
+
DEMUX
XLENAA
XLENB2

FIG. ~~6~~ 9

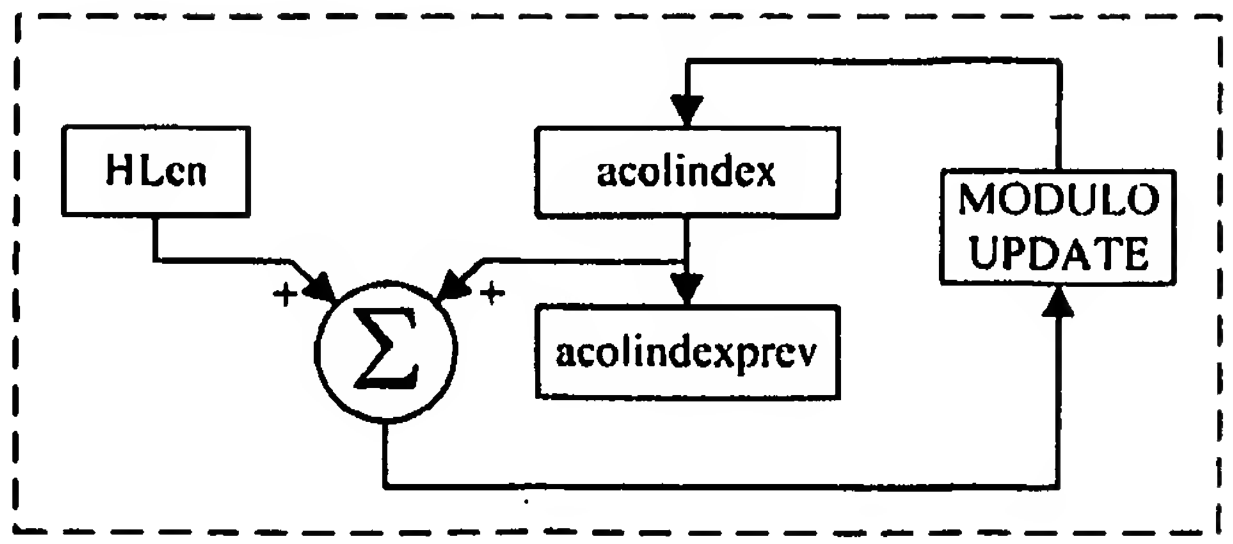


FIG. ~~8~~ 10

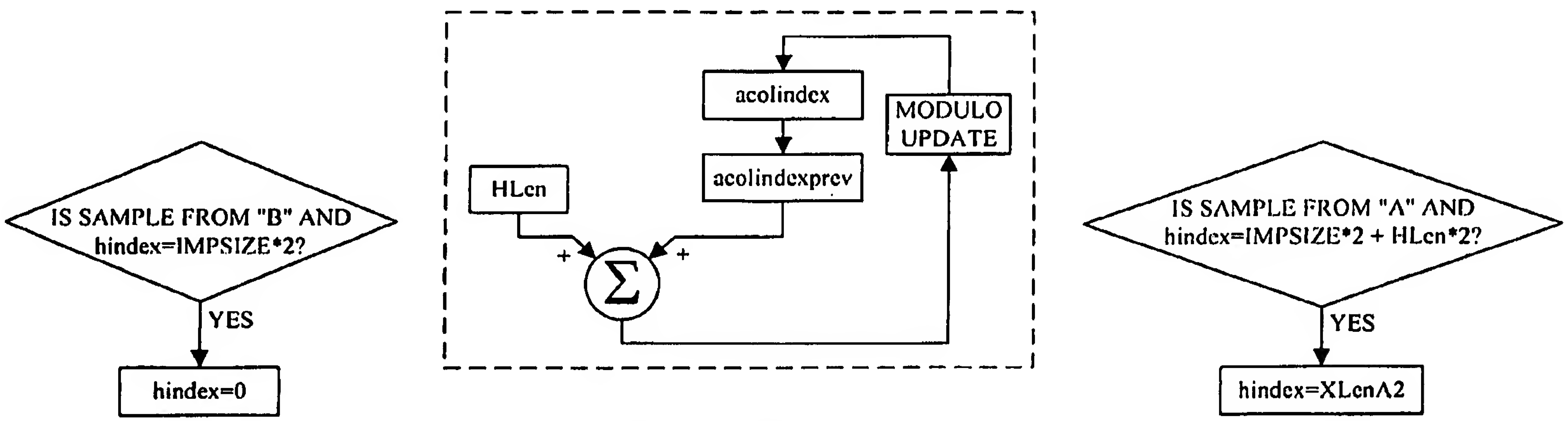


FIG. ~~9~~ 11

Tail Extension Processing

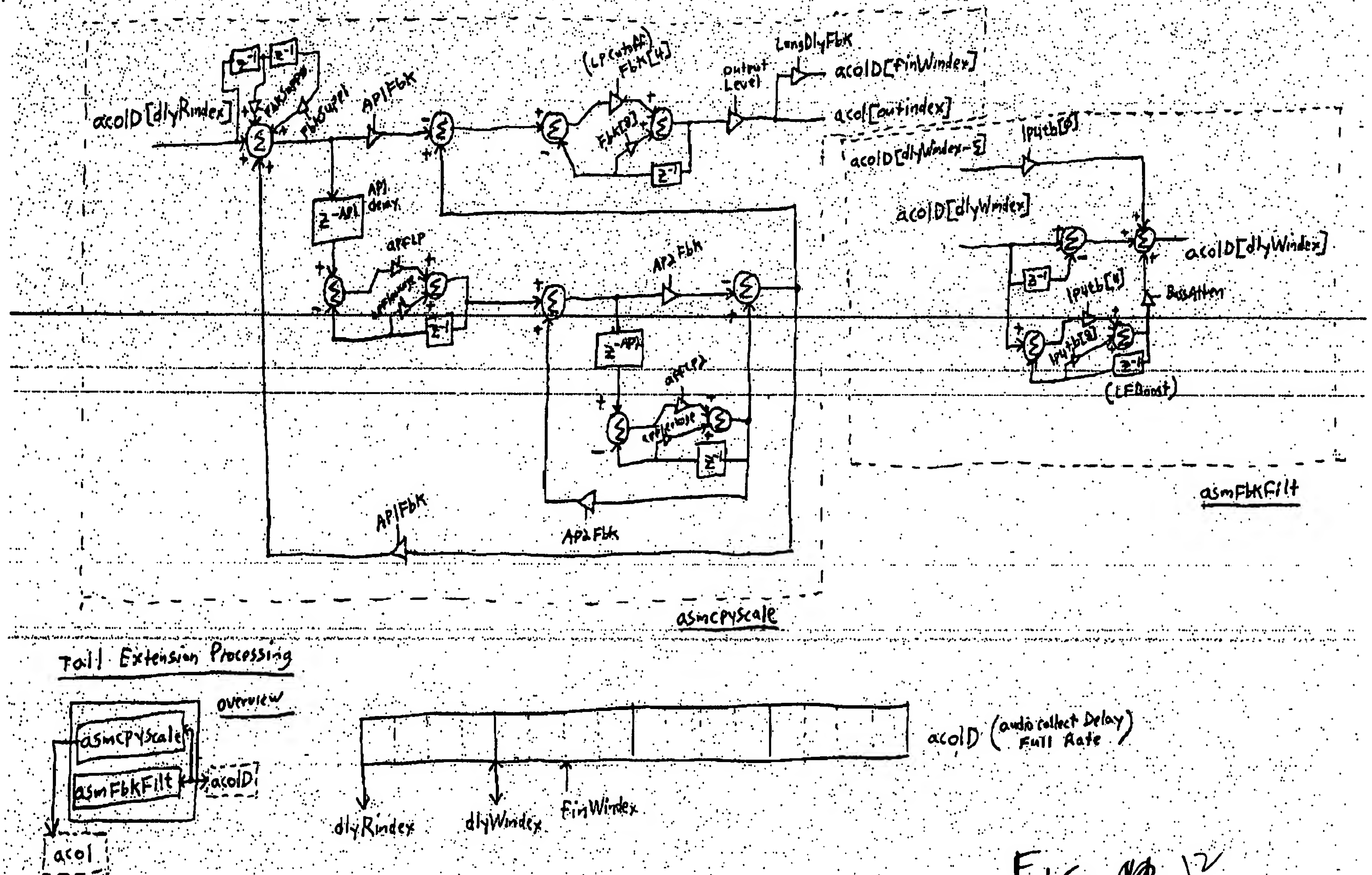


FIG 12

Document made available under the Patent Cooperation Treaty (PCT)

International application number: PCT/US04/033290

International filing date: 08 October 2004 (08.10.2004)

Document type: Certified copy of priority document

Document details: Country/Office: US
Number: 60/510,068
Filing date: 09 October 2003 (09.10.2003)

Date of receipt at the International Bureau: 22 November 2004 (22.11.2004)

Remark: Priority document submitted or transmitted to the International Bureau in compliance with Rule 17.1(a) or (b)

World Intellectual Property Organization (WIPO) - Geneva, Switzerland
Organisation Mondiale de la Propriété Intellectuelle (OMPI) - Genève, Suisse